(19) Japan Patent Office (JP)

(12) Publication of Unexamined Patent Application (A)

(11) Japanese Patent Application Laid-Open Number: 2000-13601 (P2000-13601A)

(43) Laid-Open Date: Heisei 12-1-14 (January 14, 2000)

(51) Int.Cl.[7]	Identification Code	FI		Theme Code (reference)
H04N 1/405		H04N 1/40	B	2C061
B41J 5/30		B41J 5/30	Z	2C087
29/38		29/38	Z	5B021
G06F 3/12		G06F 3/12	L	5B080
G06T 11/00		15/72	G	5C077

Request for Examination: Not requested

Number of Claims: 16 OL (17 pages in total)

(21) Application Number: Tokugan Hei 10-172384

(22) Filing Date: Heisei 10-6-19 (June 19, 1998)

(71) Applicant: 000005496
Fuji Xerox Co.,Ltd
2-17-22 Akasaka, Minato, Tokyo

(72) Inventor: Koyanagi Masahiko
c/o Fuji Xerox Co.,Ltd Green Nakai,
430 Sakai, Nakai-cho, Ashigarakami-gun,
Kanagawa

(74) Agent: 100086531
Patent Lawyer Sawada Toshio

Continued to the last page

(54)[Title of the Invention] print information processing APPARATUS, intermediate data generation APPARATUS, method of processing print information, and method of generating intermediate data

(57)[Abstract]

[Objects] To provide an apparatus and a method in each of which a gradation conversion from multi-valued gradation input data into binarized data is performed so as to achieve high-speed print processing on a printing device for printing and outputting.

[Solving Means] Disclosed is a print processing apparatus, which expands a PDL in bitmap format, and which then outputs the expanded PDL to a page printer. The print processing apparatus generates edge data on the basis of intermediate data in drawing object unit, which is generated by PDL expansion means. Then the print processing apparatus executes a gradation conversion to convert received multi-valued gradation data into binarized data, for an area defined by the edge data thus generated. The edge information of an object is generated in substantially the same resolution as that of a bitmap image, and is then directly transferred to an output means. A binarization process using a gradation conversion and drawing processes on binarized data to an output buffer are sequentially and repeatedly performed for the area defined by the edge data, and are also performed in pipeline as a sequence of processes.

[Scope of Claims]

[Claim 1] A print information processing apparatus, which generates a bitmap image based on print data from the print data described in a predetermined description language, the print data including at least one of drawing objects of character and figure, the print information processing apparatus characterized by comprising:

output means which outputs the bitmap image;

edge information generation means which directly transfers edge information of the object from the print data to the output means, the edge information being included in the print data;

a threshold matrix, which is compared with the edge information generated by the edge information generation means, and which has substantially the same resolution as that of the bitmap image;

comparing means which compares the threshold matrix with the edge information; and

a bitmap memory of the output means, in which memory the result of the comparison by the comparing means is stored.

[Claim 2] The print information processing apparatus according to claim 1, characterized in that the edge information has substantially the same resolution as that of the bitmap image.

[Claim 3] The print information processing apparatus according to claim 1, characterized by further comprising conversion means which converts the resolution of the edge information into the resolution of the matrix.

[Claim 4] A print information processing apparatus, which generates a bitmap image based on print data from the print data described in a predetermined description language, the print data including at least one of drawing objects of character and figure, the print information processing apparatus characterized by comprising:

output means which outputs the bitmap image;

edge information generation means which generates edge information and information on the kind (hereinafter referred to as kind information) of the object, from the object included in the print data;

a plurality of threshold matrices, which are compared with the edge information generated by the edge information generation means, and which are provided respectively for the kinds of object;

selecting means which selects a specific one from among the plurality of threshold matrices on the basis of kind information of the object;

comparing means which compares the threshold matrix selected by the selecting means with the edge information; and

a bitmap memory of the output means in which the result of the comparison by the comparing means is stored.

[Claim 5] An intermediate data generation apparatus, which expands print data described in a predetermined description language, the print data including at least one of drawing objects of character and figure,

the intermediate data generation apparatus characterized by comprising edge information generation means

which generates edge information and kind information of the object, from the object included in the print data.

[Claim 6] The intermediate data generation apparatus according to claim 5, characterized in that the edge information contains the kind information.

[Claim 7] The intermediate data generation apparatus according to claim 6, characterized by further comprising:

network means which is used for the print data;

means which registers information on an output device for the print data, the information including at least the resolution;

specifying means which specifies an output device for outputting the print data;

means which generates the edge information, in which at least the resolution is specified, on the basis of information on the output device specified by the specifying means; and

means which directly transfers the edge information generated by the generation means to the network.

[Claim 8] A print information processing apparatus, which generates a bitmap image based on print data described in a predetermined description language from intermediate data of the print data, the print data including at least one of drawing objects of character and figure, the print information processing apparatus characterized by comprising:

edge information recognizing means which recognizes edge information and kind information of the object, from the intermediate data of the print data;

a plurality of threshold matrices, which are compared

with the edge information recognized by the edge information recognizing means, and which are provided respectively for the kinds of object;

selecting means which selects a specific one from among the plurality of threshold matrices on the basis of the kind information of the objects, the kind information recognized by the edge information recognizing means; and

comparing means which compares the threshold matrix selected by the selecting means with the edge information.

[Claim 9] The print information processing apparatus according to claim 8, characterized by further comprising a bitmap memory of the output means in which the comparison result of the comparing means is stored.

[Claim 10] A print information processing apparatus, which receives print data described in a predetermined drawing command and, the print data including at least one of drawing objects of character and figure, and which outputs a bitmap image based on the received print data, the print information processing apparatus characterized by comprising:

edge information generation means which generates, for a drawing object included in the received print data, edge information corresponding to component line segment information on the drawing object, on the basis of the received print data;

gradation conversion means which performs a gradation conversion on an area defined by the edge information on the basis of gradation information of the drawing object included in the received print data, and which sequentially outputs

output gradation information generated by the gradation conversion to edge drawing means;

the edge drawing means, which sequentially receives the edge information generated by the edge information generation means and the output gradation information generated by the gradation conversion means, which sequentially generates bitmap images of the area defined by the edge information on the basis of the received output gradation information, and which then draws the generated bitmap images to an output buffer memory; and

print output means which prints and outputs the bitmap images stored in the output buffer memory,

the print information processing apparatus characterized in that the generation process of generating the output gradation information by the gradation conversion means, the transfer process of transferring the generated output gradation information to the edge drawing means, and the drawing process of drawing the generated bitmap images to the output buffer memory by the edge drawing means are configured to be sequentially and repeatedly performed in predetermined processing unit as a sequence of synchronized processes.

[Claim 11] The print information processing apparatus according to claim 10, characterized by further comprising intermediate information generation means which generates intermediate information based on the received print data,

the print information processing apparatus characterized in that

the intermediate information generation means generates,

with respect to one of drawing objects of character and figure included in the received print data, intermediate information, which contains gradation information unique to the drawing object and vertex information of an arbitrary number of trapezoids and triangles which approximate the shape of the contour of the drawing object, and

the edge information generation means generates the edge information with respect to each drawing object included in the received print data, on the basis of the intermediate information.

[Claim 12] The print information processing apparatus according to claim 11, characterized in that

the intermediate information generation means generates intermediate information which contains raster data having predetermined resolution and vertex information of a bounding rectangle of the raster data, with respect to a drawing object of raster included in the received print data,

the gradation conversion means performs a gradation conversion, with respect to one of drawing objects of character and figure, on the basis of gradation information of the drawing object, and

the gradation conversion means performs a gradation conversion, with respect to drawing objects of raster, by converting the raster data into the resolution of the print output means.

[Claim 13] A method of processing print information, in which a bitmap image is generated from and based on print data described in a predetermined description language, the print

data including at least one of drawing objects of character and figure, the method characterized by comprising:

an edge information generating step of generating edge information of the object included in the print data from the print data, and for directly transferring the edge information to the output means;

a comparing step of comparing the edge information generated in the edge information generating step and a threshold matrix having substantially the same resolution as that of the bitmap image;

a step of storing the result of the comparison performed in the comparing step in a bitmap memory; and

a step of outputting a bitmap image stored in the bitmap memory.

[Claim 14] A method of processing print information, in which a bitmap image is generated from and based on print data described in a predetermined description language, the print data including at least one of drawing objects of character and figure, the method characterized by comprising:

an edge information generating step of generating edge information and kind information of the object, from the object included in the print data;

a step of selecting a specific one from among a plurality of threshold matrices on the basis of the kind information of the object;

a comparing step of comparing the threshold matrix selected in the selecting step and the edge information;

a step of storing the comparison result in the comparing

step in a bitmap memory; and

a step of outputting a bitmap image stored in the bitmap memory.

[Claim 15] A method of generating intermediate data, in which print data are expanded, the print data being described in a predetermined description language, and including at least one of drawing objects of a character and a figure, the method characterized by comprising an edge information generating step of generating edge information and kind information of the object, from the object included in the print data.

[Claim 16] A method of processing print information, in which a bitmap image is generated from and based on intermediate data of print data being described in a predetermined description language, and including at least one of drawing objects of a character and a figure, the method characterized by comprising:

an edge information recognition step of recognizing edge information and kind information of the object, from the intermediate data of the print data;

a selecting step of selecting, on the basis of the kind information recognized in the edge information recognition step, a specific one from among a plurality of threshold matrices provided respectively for the kinds of object; and

a comparing step of comparing the threshold matrix selected in the selecting means and the edge information.

[Detailed Description of the Invention]

[0001]

[Technical Field to which the Invention pertains] The present invention relates to a print processing apparatus and a print processing method, in each of which data described in a predetermined drawing command are expanded in a bitmap format and then printed. In particular, the present invention relates to an apparatus and a method in each of which a process for converting received printing data represented in multi-valued gradation are converted into binarized data, and for then drawing and outputting the binarized data.

[0002]

[Prior Art] With the development of a small and fast electrophotographic page printer which is suitable for digital printing, the following print processing apparatus has become widely used. Specifically, the print processing apparatus uses a page description language (hereinafter, referred to as the PDL) with which, it is possible to freely control zooming, rotating and deforming of a drawing object by handling raster, diagrams, characters and the like in the same manner. As a result, the print processing apparatus has been significantly improved, as compared with the conventional printing system which mainly handles character information. As typical examples of a PDL of this type, known are PostScript (a trademark of Adobe Systems Incorporated), Interpress (a trademark of Xerox Corporation), and the like.

[0003] In print data written in a PDL, drawing commands representing an arbitrary drawing object in a page are arranged

in arbitrary order. For this reason, when the object represented by the print data is printed using a page printer according to the present invention, the print data must be expanded in bitmap format before printing. The bitmap expansion is the following process. First, print data are analyzed and expanded into a series of pixels each extending across the page, so that raster scan lines are formed. Then, subsequent scan lines are sequentially generated below the page.

[0004] In a CRT display device, a printing device, a digital multifunction copy machine or the like, which is digitally controlled, an image is composed of a large number of pixels each having a fixed size. In addition, the coordinates of each pixel are digitally given, and each pixel displays a binary image of white or black (or color).

[0005] There are some expensive laser printers which are capable of representing a plurality of gradations (e.g., 256 gradations) per pixel. However, an ink jet printer, which has rapidly spread, and a laser printer having a resolution of 2000 pixels per inch, which will be put to practical use, are capable of representing only a few gradations per pixel. As a method of digitally performing a gradation representation in such a binary recording device, a dither method and a density pattern method are well known as.

[0006] The dither method binarizes one pixel of a grayscale picture using a certain threshold. As shown in Fig. 13, as thresholds, a systematic dither method is commonly used, in which a dither pattern in a matrix is periodically used. Fig.

13 shows multi-valued gradation input data on the left end of the drawing, threshold data in the middle thereof, and output data on the right end thereof. In the input data, a large number of gradation value data sets exist. The gradation value data sets have various gradation values of 5, 4, 3 and 8 per pixel, respectively. These gradation values of the input pixels are compared respectively with the threshold data, i.e., threshold data sets in m rows × n columns. For example, the gradation value of input data in the first row and the first column is 5, and corresponding threshold data is 14. In this case, the relationship between the input gradation and the threshold is given as input gradation: 5 ≤ threshold: 14. Accordingly, the output data are white. Meanwhile, the gradation of input data in the second row and the second column is 6 and a corresponding threshold is 5. In this case, since the input data is larger than the threshold, the output data becomes black. In this manner, the gradation of each pixel of input data is compared with corresponding threshold data of the threshold matrix, and thereby an output is determined.

[0007] As shown in Fig. 14, in the density pattern method, one pixel of a grayscale picture is set to be a plurality of binary pixels by using a plurality of thresholds (e.g., m × m matrix). For example, suppose that the gradation of input data on the left end is 5, and that threshold data corresponding to the pixel are represented by an m × m matrix, that is a 4 × 4 matrix in this example. The gradation of the input data is compared with threshold data as in the comparison in the above-described dither method. Thereby, a value 0 or 1 is obtained, which value

corresponds to the matrix of the threshold data. Consequently, the result of output data shown on the right end of Fig. 14 is obtained.

[0008] In either method described above, a basic process for binarization is as follows. Specifically, the magnitude of one pixel of a grayscale picture having multi-valued gradation (e.g., 8 bit: 256 gradations) is compared with one threshold stored in advance, and a binary pixel is outputted on the basis of the comparison result.

[0009] In the case of a color image, the color image is decomposed into color components (e.g., CMYK: four colors of cyan, magenta, yellow, and black). Then, the above-described dither method or the density pattern method is applied to the image data of each color. Reproducibility of the image data of each color is basically governed by a gradation reproducibility of a single color. However, when the color image is reproduced, depending on how these colors overlap, moire fringes may occur due to positional errors of dots. In order to prevent moire fringes from occurring, different threshold matrices are used for the respective colors.

[0010] In a print processing apparatus for performing printing by expanding a conventional PDL in bitmap format, when a binarization process is performed, a method shown in Fig. 15 is generally employed. Specifically, a page memory having a capacity, in which multi-valued image data of one page can be stored, is prepared in advance. Then, PDL expansion processing means expands PDL data in bitmap format while temporarily retaining the PDL data as being multi-valued, and

stores the expanded PDL data in the page memory. Next, binarization means sequentially binarizes the multi-valued image of one page by using the above dither method or the like, and then generates a binary image.

[0011] Fig. 15 shows a configuration of a conventional binarization process of a multi-valued gradation data, which has been generally performed. First, PDL data are expanded by a PDL expansion means. Then, the expanded data are stored in a page memory. The data stored in the page memory are then binarized by binarization means to be thereafter outputted in bitmap format. In such a configuration as shown in Fig. 15, data to be stored in the page memory are multi-valued gradation data. In the case of data having 256 gradations per pixel, for example, gradation data of 8 bits are required for one pixel. Accordingly, the amount of data for all the pages becomes enormous. Consequently, the page memory having a large amount of storage capacity has been required. It is expected that needs for processing an image with higher density and higher resolution than ever will be increased in the future. Accordingly, it is expected that the amount of storage capacity required for the page memory would be increased if the conventional configuration were still used in the future.

[0012] In addition, Japanese Patent Application Laid-open Publication No. Hei 8-139953 discloses a system for a printing device controller which expands and outputs image data for each band in bitmap format to a printer which can output multi-valued data. Specifically, when the amount of an image data is excessive, a memory overflow may occur, or an expansion process

may not be completed in time. In such cases, in the system, output gradation is reduced, so that the system can generate and then output bitmap data for a page.

[0013]

[Problems to be Solved by the Invention] As described above, in an apparatus and a method which binarizes inputted multi-valued gradation data so as to output for printing, the multi-valued gradation data are temporarily stored in a page memory, and are thereafter binarized. Accordingly the following problems occur.

[0014] As shown in Fig. 15, in a conventional binarization processing system, a large capacity memory to store multi-valued image for one page is necessary as described above. In the future, for processing images with high density and high resolution, a large capacity memory is required so that a problem occurs that the cost is possibly increased. For example, in order to store a full-color image of A4 size having 256 gradations (8 bits) per one color for one pixel, resolution with 600 pixels per one inch requires a memory having 140 MB.

[0015] Further, the system disclosed in the above-described Japanese Patent Application Laid-open No. Hei 8-139953 is applied to a printer which enables a multi-valued output with low resolution, and has the problem that, when performing a gradation conversion, since bitmap data for one page is generated for each pixel in a software process, the speed becomes slow.

[0016] The present invention was made in view of the above-described problems and is that, in a print processing

apparatus expanding a PDL in bitmap format and outputting the same into a page printer, edge data are generated based on intermediate data in a drawing object unit generated by a PDL expansion means; a gradation conversion is performed from inputted multi-valued gradation data into binarized data with respect to an area defined by the edge data thus generated; and the binarized data thus converted are sequentially store in an output buffer. The present invention has a configuration in which these steps are sequentially repeatedly performed with respect to areas defined by the edge data, and a gradation conversion process and a drawing process to an output buffer are performed in pipeline as a sequence of processes. Accordingly, the present invention is capable of providing a print processing apparatus and a print processing method in which, for example, even when an output buffer stores data for one page, data for pixels are binary data so that a large capacity page memory or a band memory is not required unlike a conventional print processing apparatus which temporarily stores multi-valued image data, e.g., data of 8 bit/pixel, and thus a fast print at low cost can be achieved.

[0017]

[Means for Solving the Problems] The present invention has been made for the purpose of achieving the above-described object. A print information processing apparatus of the present invention is a print information processing apparatus, which generates a bitmap image based on print data from the print data described in a predetermined description language, the print data including at least one of drawing objects of a

character and a figure. The print information processing apparatus is characterized by including: output means which outputs the bitmap image; edge information generation means which directly transfers edge information of the object from the print data to the output means, the edge information being included in the print data; a threshold matrix, which is compared with the edge information generated by the edge information generation means, and which has substantially the same resolution as that of the bitmap image; comparing means which compares the threshold matrix with the edge information; and a bitmap memory of the output means, in which memory the result of the comparison by the comparing means is stored.

[0018] In addition, the print information processing apparatus of the present invention is characterized in that the edge information has substantially the same resolution as that of the bitmap image.

[0019] Moreover, the print information processing apparatus of the present invention is characterized by further including conversion means which converts the resolution of the edge information into the resolution of the matrix.

[0020] Furthermore, a print information processing apparatus of the present invention is a print information processing apparatus, which generates a bitmap image based on print data from the print data described in a predetermined description language, the print data including at least one of drawing objects of a character and a figure. The print information processing apparatus is characterized by including: output means which outputs the bitmap image; edge information

generation means which generates edge information and kind information of the object, from the object included in the print data; a plurality of threshold matrices, which are compared with the edge information generated by the edge information generation means, and which are provided respectively for the kinds of object; selecting means which selects a specific one from among the plurality of threshold matrices on the basis of kind information of the object; comparing means which compares the threshold matrix selected by the selecting means with the edge information; and a bitmap memory of the output means in which the result of the comparison by the comparing means is stored.

[0021] In addition, an intermediate data generation apparatus of the present invention is an intermediate data generation apparatus, which expands print data described in a predetermined description language, the print data including at least one of drawing objects of a character and a figure. The intermediate data generation apparatus is characterized by including the intermediate data generation apparatus characterized by comprising edge information generation means which generates edge information and kind information of the object, from the object included in the print data.

[0022] Moreover, the intermediate data generation apparatus of the present invention is characterized in that the edge information contains the kind information.

[0023] Furthermore, the intermediate data generation apparatus of the present invention is characterized by further including: network means which is used for the print data; means

which registers information on an output device for the print data, the information including at least the resolution; specifying means which specifies an output device for outputting the print data; means which generates the edge information, in which at least the resolution is specified, on the basis of information on the output device specified by the specifying means; and means which directly transfers the edge information generated by the generation means to the network.

[0024] In addition, a print information processing apparatus of the present invention is a print information processing apparatus, which generates a bitmap image based on print data described in a predetermined description language from intermediate data of the print data, the print data including at least one of drawing objects of a character and a figure. The print information processing apparatus is characterized by including: edge information recognizing means which recognizes edge information and kind information of the object, from the intermediate data of the print data; a plurality of threshold matrices, which are compared with the edge information recognized by the edge information recognizing means, and which are provided depending on the kinds of object; selecting means which selects a specific one from among the plurality of threshold matrices on the basis of the kind information of the objects, the kind information recognized by the edge information recognizing means; and comparing means which compares the threshold matrix selected by the selecting means with the edge information.

[0025] Moreover, the print information processing apparatus of the present invention is characterized by further including a bitmap memory of the output means in which the comparison result of the comparing means is stored.

[0026] In addition, a print information processing apparatus of the present invention is a print information processing apparatus, which receives print data described in a predetermined drawing command and, the print data including at least one of drawing objects of a character and a figure, and which outputs a bitmap image based on the received print data. The print information processing apparatus is characterized by including: edge information generation means which generates, for a drawing object included in the received print data, edge information corresponding to component line segment information on the drawing object, on the basis of the received print data; gradation conversion means which performs a gradation conversion on an area defined by the edge information on the basis of gradation information of the drawing object included in the received print data, and which sequentially outputs output gradation information generated by the gradation conversion to edge drawing means; the edge drawing means, which sequentially receives the edge information generated by the edge information generation means and the output gradation information generated by the gradation conversion means, which sequentially generates bitmap images of the area defined by the edge information on the basis of the received output gradation information, and which then draws the generated bitmap images to an output buffer

memory; and print output means which prints and outputs the bitmap images stored in the output buffer memory. The print information processing apparatus is also characterized in that the generation process of generating the output gradation information by the gradation conversion means, the transfer process of transferring the generated output gradation information to the edge drawing means, and the drawing process of drawing the generated bitmap images to the output buffer memory by the edge drawing means are configured to be sequentially and repeatedly performed in predetermined processing unit as a sequence of synchronized processes.

[0027] Moreover, the print information processing apparatus of the present invention is characterized by further including intermediate information generation means which generates intermediate information based on the received print data. The print information processing apparatus is also characterized in that the intermediate information generation means generates, with respect to one of drawing objects of character and figure included in the received print data, intermediate information, which contains gradation information unique to the drawing object and vertex information of an arbitrary number of trapezoids and triangles which approximate the shape of the contour of the drawing object, and that the edge information generation means generates the edge information with respect to each drawing object included in the received print data, on the basis of the intermediate information.

[0028] Furthermore, the print information processing

apparatus of the present invention is also characterized in that the intermediate information generation means generates intermediate information which contains raster data having predetermined resolution and vertex information of a bounding rectangle of the raster data, with respect to a drawing object of raster included in the received print data, that the gradation conversion means performs a gradation conversion, with respect to one of drawing objects of character and figure, on the basis of gradation information of each drawing object, and that the gradation conversion means performs a gradation conversion, with respect to drawing objects of raster, by converting the raster data into the resolution of the print output means.

[0029] In addition, a method of processing print information of the present invention is a method, in which a bitmap image is generated from and based on print data described in a predetermined description language, the print data including at least one of drawing objects of a character and a figure. The method is characterized by including: an edge information generating step of generating edge information of the object included in the print data from the print data, and for directly transferring the edge information to the output means; a comparing step of comparing the edge information generated in the edge information generating step and a threshold matrix having substantially the same resolution as that of the bitmap image; a step of storing the result of the comparison performed in the comparing step in a bitmap memory; and a step of outputting a bitmap image stored in the bitmap memory.

[0030] In addition, a method of processing print information of the present invention is a method, in which a bitmap image is generated from and based on print data described in a predetermined description language, the print data including at least one of drawing objects of a character and a figure. The method is characterized by including: an edge information generating step of generating edge information and kind information of the object, from the object included in the print data; a step of selecting a specific one from among a plurality of threshold matrices on the basis of the kind information of the object; a comparing step of comparing the threshold matrix selected in the selecting step and the edge information; a step of storing the comparison result in the comparing step in a bitmap memory; and a step of outputting a bitmap image stored in the bitmap memory.

[0031] In addition, a method of generating intermediate data of the present invention is a method, in which print data are expanded, the print data being described in a predetermined description language, and including at least one of drawing objects of a character and a figure. The method is characterized by including an edge information generating step of generating edge information and kind information of the object, from the object included in the print data.

[0032] In addition, a method of processing print information of the present invention is a method, in which a bitmap image is generated from and based on intermediate data of print data being described in a predetermined description language, and including at least one of drawing objects of a character and

a figure. The method is characterized by including: an edge information recognition step of recognizing edge information and kind information of the object, from the intermediate data of the print data; a selecting step of selecting, on the basis of the kind information recognized in the edge information recognition step, a specific one from among a plurality of threshold matrices provided respectively for the kinds of object; and a comparing step of comparing the threshold matrix selected in the selecting means and the edge information.

[0033]

[Embodiment Mode of the Invention] Fig. 1 is a view showing a basic configuration of a print processing apparatus of the present invention. In Fig. 1, reference numeral 1 denotes print data, which include at least a drawing object of one of a character object, a figure object, and a raster object, and which are described in predetermined drawing commands; reference numeral 2 denotes edge data generation means, which receives the print data, which then interprets the received print data, and which thereby generates data on a filling edge used with which the drawing object is filled in parallel to scanning lines; reference numeral 3 denotes binarization means, which binarizes gradation data of the drawing object that is inputted in multi-valued form by using a predetermined threshold so as to generate bitmap data, and which then outputs the generated data in synchronization with a process in edge drawing means 4; reference numeral 4 denotes the edge drawing means, which receives coordinates data and the binarized bitmap data, and which draws a bitmap image of each edge unit;

and reference numeral 5 denotes print output means which prints and outputs the generated bitmap image.

[0034] It is preferred that at least the binarization means 3 and the edge drawing means 4 be configured of dedicated hardware, among the above-described components. By performing a binarization process and an edge drawing process using the above-described means configured of hardware, fast processes are made possible. In addition, upon receipt of edge data generated by the edge data generation means 2, the binarization means 3 and the edge drawing means 4 perform the following processes, respectively. Specifically, the binarization means 3 sequentially performs the binarization process on each area defined by the edge data; and the edge drawing means 4 draws the binarized data in an output buffer. That is, since the synchronized processes are performed for each scanning line in an edge unit, it is not necessary to have a large capacity page buffer memory which has been heretofore required for storing multi-valued gradation image data. Accordingly, the application of the present invention to the print processing apparatus enables the print processing apparatus to perform a binarization process on the inputted multi-valued gradation print data and perform a bitmap expansion, so that fast print processing can be achieved at low cost.

[0035]

[Embodiment] An embodiment of a print processing apparatus of the present invention is described below with reference to the accompanying drawings. Fig. 2 is a block diagram showing a

basic configuration of an embodiment of print processing apparatus of the present invention.

[0036] An overview and operation of each component of the embodiment are described below. Print data 1 are described in the PDL, which can be processed in a print processing apparatus. In addition, the print data 1 are generated from document data created using an application program that processes a creation and an edition and the like of a document, and that is executed on an unillustrated personal computer or workstation.

[0037] The PDL handled in the embodiment is, for example, PostScript, but may be PDF (Portable Document Format), GDI, or the like.

[0038] An print data spool unit 11 includes a communication function for receiving the print data 1; a memory function for temporarily storing the print data 1 until the data are outputted to a print data interpretation unit 12; and the like.

[0039] The print data interpretation unit 12 cuts out, as a token, the print data 1 inputted from the print data spool unit 11 according to predetermined syntaxes of a PDL. Thereafter, the print data interpretation unit 12 interprets the token, and then converts the token into internal commands and arguments of the internal commands. The internal commands include a drawing command, a drawing state command and the like. By the drawing command, a character, a figure, or raster is drawn. By the drawing state command, information necessary for the drawing, such as color and line attributes is set. The internal command generated by the print data interpretation

unit 12 is transferred to an intermediate data generation unit 13.

[0040] The intermediate data generation unit 13 generates intermediate data of a format to be described later, according to the drawing command of character, figure or raster. In addition, the intermediate data generation unit 13 divides the generated intermediate data for each band so that blocks in subsequent stages can process the band units, into which one page is divided by a predetermined number. The size of each band unit is determined depending on the size of an output data storing unit 20. The intermediate data that have been divided into the band units are stored in an intermediate data storing unit 14.

[0041] These processes from the above-described print data interpretation unit 12 to the intermediate data generation unit 13 are performed on all of the drawing objects constituting a page. Thereby, intermediate data having divided for band unit for each page are stored in the intermediate data storing unit 14.

[0042] An intermediate data input unit 15 sequentially reads intermediate data from the intermediate data storing unit 14. The intermediate data input unit 15 partly sorts the data as necessary, and then transfers the data to an edge data generation unit 16.

[0043] The edge data generation unit 16 generates, from the intermediate data, edge data of a plurality of filling edges, with which drawing objects are filled in parallel to scanning lines. The edge data are composed of the coordinates of a start

point and an end point of the filling edge and gradation data for the filling. The gradation data are configured of multi-valued gradation data, which can have a plurality of values for one pixel. When the drawing object is of character or figure, the gradation data take a fixed value, i.e., represent the same gradation, on all of the pixels of the drawing object. Meanwhile, when the drawing object is raster, gradation data having different values for respective pixels are inputted as raster gradation data. Edge data generated in the edge data generation unit 16 can be temporarily stored in an unillustrated edge data buffer memory. Then, the stored edge data are sequentially supplied from the edge data buffer memory to a binarization processing unit.

[0044] The binarization processing unit 17 sequentially performs a binarization process on areas defined respectively by the edge data. The inputted multi-valued gradation data of an area defined by the edge data are compared with predetermined threshold data. Then, the multi-valued gradation data are binarized, on the basis of the obtained result from the comparison. As an algorithm for converting multi-valued gradation data on the basis of binary data, a dither method or a density pattern method is used. The bitmap data that have been binarized in the binarization processing unit are transferred to an edge drawing unit 19 in synchronization with a drawing process for an output data storing unit in the edge drawing unit 19.

[0045] That is, the process of generating binary data by the binarization processing unit 17, the process of transferring

the binary data thus generated to the edge drawing unit 19, and the process of drawing by the edge drawing unit 19 to an output data storing unit 20 are repeatedly performed in a predetermined processing unit, e.g., 32 bit processing word unit.

[0046] A threshold data storing unit 18 is a memory in which threshold matrix data (refer to Figs. 5, 13 and 14) used for binarization are stored. When data are outputted to a color printing device, different threshold matrices are used respectively for colors of CMYK (cyan, magenta, yellow and black). Moreover, different threshold matrices may be used respectively for drawing objects.

[0047] The edge drawing unit 19 calculates a corresponding memory address in the output data storing unit 20 from edge data generated in the edge generation unit 16. Then, the edge drawing unit 19 performs writing the bitmap data for a plurality of pixels all at once, which data have been generated by the binarization unit 17, to the calculated address.

[0048] The output data storing unit 20 is configured of two band buffer memories. Specifically, while a drawing process is performed by the edge drawing unit 19 using one of the two buffers, output bitmap data are transferred to a printing device 22 through a printing device control unit 21 using the other one of the two buffers.

[0049] The printing device control unit 21 transfers, to the printing device 22, bitmap data for one band unit stored in the output data storing unit 20 in response to an output timing of the printing device 22. Concurrently, the printing device

control unit 21 controls the state of the printing device 22, and manages the printing device 22.

[0050] The printing device 22 receives bitmap data outputted from the output data storing unit 20 on the basis of the control by the printing device control unit 21, and then prints the bitmap data on a printing sheet of paper. Precisely, the printing device 22 is a color page printer using an electrophotographic process of a laser scanning system. The color page printer is capable of outputting a color image by repeatedly performing exposure, development and transfer for each color of C, M, Y and K.

[0051] As a mounting mode of the embodiment, units ranging from the print data spool unit 11 to the intermediate data storing unit 14 are preferably mounted on a host computer such as a personal computer or a workstation, which includes an MPU, a physical memory, a secondary storage device and the like. On the other hand, units ranging from the intermediate data input unit 15 to the printing device control unit 21 are preferably mounted on a card-like board, which is to be attached to the printing device 22 as dedicated hardware. Moreover, the host computer is preferably connected to the dedicated hardware with a high-speed serial bus such as IEEE 1394.

[0052] Next, an intermediate data format of the embodiment is described. Intermediate data includes information on the contours of a drawing object and gradation data for filling the object. Contour data of a character or a figure to be used is obtained as follows. Specifically, an arbitrary polygon, which is obtained by approximating all the contour of a drawing

object by using straight lines, is first generated. Then, the polygon is divided into trapezoids and triangles with scanning lines passing through the vertexes of the polygon. Then, the results of the division are used for the contour data.

[0053] Fig. 3 is a conceptual diagram showing the contour data of characters or figures. A polygon t0 indicated by a drawing command of print data is first divided into a triangle t1, a trapezoid t2, triangles t3 and t4 by scanning lines passing through the vertexes of the polygon t0. Among those, a polygon which overlaps a boundary of the band units is further divided by the overlapped boundary. Consequently, the polygon t0 is divided into eight drawing objects t5 to t12. Each trapezoid is represented by the coordinates of four vertexes (lux, rux, uy, llx, rlx, ly).

[0054] Gradation data of a character or a figure are assumed to have a value (256 gradations) of 8 bits for each of colors CMYK. The gradation value of a character or a figure is the same for each drawing object. Contour data of raster are coordinate values of the vertexes of a bounding rectangle. Gradation data of raster are raster data of 8 bits for each of colors CMYK. Values of raster data are different between pixels.

[0055] Next, edge data generated from intermediate data of the embodiment is described. Fig. 4 is a conceptual diagram showing how edge data are generated from intermediate data.

[0056] In the edge data generation unit 16, two Digital Differential Analyzers (DDAs) are executed in order to calculate coordinate points of both of the left and right sides

of a trapezoid. On the left side, the DDA is executed from (lux, uy) toward (llx, ly). On the right side, the DDA is executed from (rux, uy) toward (rlx, ly). As a result, for each scanning line y, the start coordinates sx and the end coordinates ex of an edge for filling the inside of the trapezoid are calculated. These coordinates (sx, ex) are paired with the corresponding gradation data, and outputted to the subsequent stages, i.e., the binarization processing unit 17 and the like.

[0057] Next, the binarization processing unit 17 and the threshold data storing unit 18 are described in detail. Fig. 5 is a view showing a state in which threshold matrix data M of the size $m \times m$ are stored in the threshold data storing unit 18.

[0058] In an address A of the threshold data storing unit 18, threshold M(0, 0) is stored, and in an address A+1, threshold M(1, 0) is stored. The same processes are repeated until all the thresholds are stored. In an example of Fig. 5, predetermined thresholds are stored respectively in M(0, 0) to M(m-1, m-1). In addition, each threshold M(i, i) has a value of 8 bits. Gradation data included in input data are compared with the threshold data, and an output value binarized based on the magnitude thereof is determined.

[0059] Fig. 6 is a block diagram showing an internal configuration of the binarization processing unit 17. A bit shaping unit 171 receives gradation data of a drawing object to be processed, and expands or divides the gradation data into

a unit (for example, 256 bits) to be processed by a comparing unit 173, and then the gradation data are outputted.

[0060] In the case of a character or a figure, for each drawing object, 32 sets of gradation data, each inputted in 8 bits for each color, are arranged in parallel, and are then outputted as data of 256 bits. In the case of raster, when the size of raster data is larger than 32 pixels, the data are divided into data pieces of 256 bits for each color, and thereafter outputted.

[0061] A memory control unit 172 receives the coordinate data (sx, ex, y) of each edge, and calculates an address in which a threshold corresponding to the coordinate data is stored. Then the memory control unit 172 reads a corresponding threshold from threshold data stored in the threshold data storing unit 18. Thereafter, the memory control unit 172 outputs data of 32 sets, i.e., data of the unit of 256 bits, to the comparing unit 173.

[0062] The comparing unit 173 includes 32 sets of 8 bit magnitude comparators. The comparing unit 173 compares gradation data to be inputted and threshold data, and then outputs a compared result to the edge drawing unit 19 as a filling pattern. The comparison of this case may be performed according to the above-described dither method shown in Fig. 13, or according to the density pattern method shown in Fig. 14. The comparison result for each pixel is 1 (output in black or color) when gradation value > threshold; or 0 (white) when gradation value ≤ threshold. A filling pattern is generated

in the unit of 32 pixels in synchronization with the drawing process of the edge drawing unit 19.

[0063] Next, a binarization process in the case of using a threshold matrix different for each drawing object is described.

[0064] Fig. 16 is a view showing a process in which bitmap data are generated from print data through intermediate data 100 and then edge data 200. Since the intermediate data generation unit 13, the edge data generation unit 16, and the binarization processing unit 17 have already been described, descriptions thereof will be omitted. In the threshold data storing unit 18, the following threshold matrices are stored: a threshold matrix FC for character object (cyan); a threshold matrix FM for character object (magenta); a threshold matrix FY for character object (yellow); a threshold matrix FK for character object (black); a threshold matrix GC for figure object (cyan); a threshold matrix GM for figure object (magenta); a threshold matrix GY for figure object (yellow); a threshold matrix GK for figure object (black); a threshold matrix RC for raster object (cyan); a threshold matrix RM for raster object (magenta); a threshold matrix RY for raster object (yellow); and a threshold matrix RK for raster object (black).

[0065] Examples of data formats respectively of the intermediate data 100 generated by the intermediate data generation unit 13 and the edge data 200 generated by the edge data generation unit 16 are shown in Fig. 17. Fig. 17 extracts and shows parts corresponding to one drawing object. In each of ObjectID fields 101 and 201, a number indicating the kind

(one of a character, a figure and raster) of the drawing object is stored. In the field 201, the same value as that in the field 101 is set by the edge data generation unit 16. When the drawing object is of character or of figure, gradation values (8 bits each) for respective colors of CMYK are stored in each of Color fields 102 and 202. When the drawing object is raster, a pointer (a memory address) to a memory in which the raster data is stored is stored in each of the Color fields 102 and 202. In the field 202, the same value as that in the field 102 is set by the edge data generation unit 16. In each TRData field 103, a set of the contour data of a trapezoid described in Fig. 4 is stored. Each set of the contour data is converted into edge data by the edge data generation unit 16 by using the method described in Fig. 4, and the converted edge data are stored in a corresponding EdgeData field 203.

[0066] The flow of the binarization process in the case where yellow is processed in the three kinds of drawing object: a character; a figure; and raster is described. First, the case of a character object is described. Once a character object is inputted into the binarization processing unit 17 (corresponding to Step S21), the binarization processing unit 17 checks the ObjectID field 201, and determines that the inputted drawing object is a character. Next, the binarization processing unit 17 takes out a gradation value for yellow from the Color field 202 (corresponding to Step S31). From the threshold data storing unit 18, the binarization processing unit 17 receives threshold data of the threshold matrix FY for character object (yellow), which threshold data

correspond to an area for drawing (Step S32). The binarization processing unit 17 performs a comparison process by using the inputted gradation value and the threshold (Step S33), and then outputs bitmap data of the comparison result (Step S34).

[0067] Next, the case of a figure object is described. Once a figure object is inputted into the binarization processing unit 17 (corresponding to Step S21), the binarization processing unit 17 checks the ObjectID field 201, and determines that the inputted drawing object is a figure. Next, the binarization processing unit 17 takes out a gradation value for yellow from the Color field 202 (corresponding to Step S31). From the threshold data storing unit 18, the binarization processing unit 17 receives threshold data of the threshold matrix GY for figure object (yellow), which threshold data correspond to an area for drawing (Step S32). The binarization processing unit 17 performs the comparison process by using the inputted gradation value and the threshold (Step S33), and then outputs bitmap data of the comparison result (Step S34).

[0068] Last, the case of a raster object is described. Once a raster object is inputted into the binarization processing unit 17 (corresponding to Step S21), the binarization processing unit 17 checks the ObjectID field 201, and determines that the inputted drawing object is a raster object. Next, the binarization processing unit 17 takes out a address for raster data from the Color field 202, and reads the raster data from the memory address (corresponding to Step S31). From the threshold data storing unit 18, the binarization processing unit 17 receives threshold data of the threshold

matrix RY for raster object (yellow), which threshold data correspond to an area for drawing, are inputted (Step S32). The binarization processing unit 17A performs the comparison process by using the inputted gradation value and the threshold (Step S33), and then outputs bitmap data of the comparison result (Step S34). Incidentally, details of the comparing process are referred to description on Figs. 5, 6, 10, 13 and 14.

[0069] Next, the flow of the entire processes is described with reference to Figs. 7 to 10. Fig. 7 is a flowchart showing the flow of processes in which intermediate data are generated from the print data 1.

[0070] In Step S1, once the print data 1 is inputted into the print data spool unit 11, the print data interpretation unit 12 sequentially performs a lexical analysis, a token cutout, a token interpretation and the like in Step S2.

[0071] As a result of the interpretation, when input data in Step S3 represent a drawing command, the process goes to Step S6, and the intermediate data generation unit 13 generates intermediate data corresponding to the drawing object. As described above, the intermediate data includes information on the contour of the drawing object and gradation data for filling the inside of the drawing object. When the input data does not represent a drawing command, the process goes to Step S4. When the input data are determined to be a drawing state command in Step S4, the process goes to Step S5. In Step S5, an attribute and the like required for drawing are set. After the process in Step S5 is executed, the process returns to Step

S2. When the input data is determined not to be a drawing state command in Step S4, a process corresponding to the command is performed, and thereafter the process returns to Step S2.

[0072] In Step S6, the intermediate data generation unit 13 generates intermediate data of the drawing object instructed by the drawing command. Thereafter, in Step S7, it is checked whether or not all the drawing objects for one page have been processed. When there is still a command having not been processed, the process returns to Step S2 and the same process is repeated.

[0073] Once all the intermediate data for one page have been generated, the process goes to Step S8. Then, the intermediate data are divided for band unit as described using Fig. 3, and then outputted to the intermediate data storing unit 14. Thereby, intermediate data generation processes for one page is completed.

[0074] Fig. 8 is a flowchart showing the flow of processes in which edge data are generated from intermediate data. In Step S11, the intermediate data input unit 15 fetches intermediate data for each word from the intermediate data storing unit 14 and decodes the fetched intermediate data. Contour data and gradation data included in the inputted intermediate data are passed to the edge data generation unit 16 in a drawing object unit.

[0075] Next, in Step S12, initial values for a DDA process are calculated on the basis of the coordinate data of the vertices of a trapezoid or a triangle, such as shown for example in Fig. 4. The trapezoid or the triangle is represented by contour

data included in the intermediate data. The contents of the initial values are the coordinate values (sx, ex, y) of the start point of the DDA, the number of crossing with scanning lines dy=ly-uy, the slope of the left side dlx=(llx-lux)/dy, and the slope of the right side drx=(rlx-rux)/dy. Subsequently, in Step S13, the first edge data (sx, ex, y)=(lux, rux, uy) are outputted to the binarization processing unit 17. It is also possible to employ a configuration in which the generated edge data are temporarily stored in an edge data buffer before being outputted to the binarization processing unit 17.

[0076] In Step S14, in order to confirm that the processing on the object has been completed, it is checked whether or not dy=0. When it is true, the edge list generation process on the drawing object is terminated, and the process goes to Step S16. When it is false, the process goes to Step S15.

[0077] In Step S15, the DDA process is performed, and edge data for the next scanning line are calculated. To be specific, four processes: sx=sx+dlx; ex=ex+drx; y=y+1; and dy=dy-1 are performed in parallel by hardware. When the calculation of DDA is completed, the process returns to Step S13. Then, upon receipt of a data request from the binarization processing unit 17, the same process is again performed.

[0078] In Step S16, when there is still an unprocessed drawing object left in a band, the process returns to Step S11, and the edge data generation process is performed on the unprocessed drawing object in Steps S11 to S16. This process is performed on all the drawing objects in the band. When edge

data for all the drawing objects in the band are generated, the process is terminated.

[0079] Fig. 9 is a flowchart showing the flow of processes in which bitmap image data in object unit are generated from edge data of each object.

[0080] In Step S21, once the initial edge data composing an object are inputted into the binarization processing unit 17, a binarization process is performed in Step S22. Thereby, a filling pattern (bitmap data) for one word (32 bits) is generated from gradation data of the edge.

[0081] In Step S23, the bitmap data thus generated are transferred from the binarization processing unit 17 to the edge drawing unit 19.

[0082] Next, in Step S24, the edge drawing unit 19 calculates a memory address, in which the edge is written, and a word mask, from the coordinate data of the edge. Then, the edge drawing unit 19 accesses to the output data storing unit 20, and draws the bitmap data. At this time, as needed, the edge drawing unit 19 reads base bitmap data before writing. Then the edge drawing unit 19 obtains a product by performing a logical AND operation on the base bitmap data and one obtained by reversing a word mask. Thereafter, the following process is performed. Specifically, a logical ADD operation is performed on a product, which is obtained by a logical AND operation of a pattern and a word mask, and the product of the above logical AND operation. The result of the logical ADD operation is then drawn to the output data storing unit 20.

[0083] In Step S25, it is checked whether the edge has been entirely drawn from the start point to the end point. When there still is a word left without being drawn in the edge, the process returns to Step S22 to repeat the above processes. When all the words in the edge are drawn, the edge drawing processes for the edge are completed.

[0084] In the binarization processing unit 17, filling patterns (bitmap data), each of which is binary data for one word (32 bits), are generated all together from the gradation data of the edge. The generated bitmap data are then transferred from the binarization processing unit 17 to the edge drawing unit 19. The edge drawing unit 19 accesses for the bitmap data all together to the output data storing unit 20 in each word unit (32 bits), and draws the bitmap data to the output data storing unit 20.

[0085] The series of the processes, i.e., the processes of generating binary data by the binarization processing unit 17, the processes of transferring the generated binary data to the edge drawing unit 19, and the processes of drawing the bitmap data to the output data storing unit 20 by the edge drawing unit 19, are successively and repeatedly performed. When the processes are performed on a page-by-page basis, edge data composing a drawing object at a lowermost level among a plurality of drawing objects included in a band to be processed are first inputted. Then, the binarization process, the transferring, and the bitmap drawing process in one word unit are repeated so that a bitmap drawing of one object is completed. Moreover, edge data on the next drawing object are inputted

into the binarization processing unit 17, and the binarization process, the transferring, and the bitmap drawing process are performed on the basis of the inputted edge data. These processes are sequentially performed on drawing objects in the band.

[0086] Fig. 10 is a flowchart showing the flow of the binarization process. In Step S31, gradation data of a drawing object are inputted to the binarization processing unit 17. On a character or a figure, it suffices that gradation data of 8 bits for one drawing object are inputted only once. However, on raster, raster gradation data are inputted one after another until all the edges are drawn. The inputted gradation data are expanded or divided in 256 bit unit by the bit shaping unit 171, and outputted to the comparing unit 173.

[0087] In Step S32, the memory control unit 172 reads thresholds corresponding to coordinate data of the respective edges one after another from the threshold data storing unit 18. The thresholds thus read, which are in 32 sets = 256 bits, are collectively transferred to the comparing unit 173.

[0088] In Step S33, the comparing unit 173 performs a comparing process on the inputted gradation data and the read threshold data. A filling pattern of 32 bits obtained as the comparison result is outputted to the edge drawing unit 19 in Step S34.

[0089] The flow described above is the flow of the binarization process for one word. In the case of a character or a figure, Steps S32 to S34 are repeated until all the edges are processed. In the case of raster, Steps S31 to S34 are repeated until all the edges are processed.

[0090] In the above-described embodiment, edge data are generated separately for each object. Then, binarization process is performed on each object. Thereafter, the results are outputted in bitmap. Accordingly, when objects overlap each other, and concurrently when a portion where the overlap occurs is not transparent, edge data are taken out from the objects in order from the lower level to the upper level, and the binarization process and the bitmap output are performed for each object. As a result, bitmap in the portion where the overlap occurs is overwritten on the output memory.

[0091] For example, as shown in Fig. 11, suppose that an object A and an object B partially overlap each other, and a part of the object A is hidden under the object b. In this case, first, the binarization process is performed on the object A on the basis of edge data of the object A, so that bitmap data of the object A is expanded in a memory. Thereafter, the binarization process is performed on the object B on the basis of edge data of the object B, so that bitmap data of the object B is expanded in the memory. Thereafter, in the overlapped part, the bitmap data of the object A are overwritten with the bitmap data of the object B.

[0092] In this case, binarized data of the object at the lower level, on which the overlap occurs, are not used in an actual output. This makes it virtually pointless to perform the binarization process on a part of the object at the lower level, which part is hidden under the object at the upper level. Fig. 12 shows a flow configured for avoiding such an unnecessary binarization process, in which the presence of overlapping is

determined so that the binarization process is performed only on a part necessary for an output.

[0093] In Step S41 of Fig. 12, the presence of overlapping is determined. This process is performed by comparing edge data of all the objects included in a band, which data are obtained through the edge data generation process shown in Fig. 8, for example.

[0094] On an object on which an overlapping part is detected by performing the determination of overlapping on objects in Step S41, a correction process is performed on edge data in Step S42. That is, for example, the contour A1A2A3A4 of the object A shown in Fig. 14 is set as another contour A1A2A3C2B2C1 where the overlap is taken into consideration. Thereby, edge data of the object A are corrected so that the edge data based on the contour A1A2A3C2B2C1 can be composed.

[0095] The corrected edge data are inputted to binarization means (Step S43), and a binarization process is performed on the corrected edge data(Step S44). These Steps S43 and S44 correspond to Steps S21 and S22 of Fig. 9, and thereafter, Step S23 and subsequent Steps shown in Fig. 9 are performed.

[0096] According to the above-described configuration, it is possible to omit a binarization process, i.e., a comparing process on input data and a threshold and an output process of the comparison result, which are supposed to be performed on an overlapping part of one at the lower level of objects overlapping each other.

[0097] Further, the following configuration is possible. That is, when objects overlap each other or when objects are

connected to each other, gradation data for the objects are compared with each other. When the objects including an overlapping part or a connecting part have the same gradation, the objects are merged. Then by treating the objects as a single object, corrected edge data are generated. On the basis of the edge data thus generated, the binarization process is performed. For example, when the two objects A and B shown in Fig. 11 have the same gradation, it is possible to perform the binarization process as follows. Specifically, by treating these two objects as one object with the vertexes A1A2A3C2B3B4B1C1, an edge data correction is performed so that edge data of this one object are generated. Accordingly, a binarization process, which is performed separately on each of the two objects in different steps, can be performed on the merged object as the binarization process in one step.

[0098] The above-described embodiments of the print processing apparatus and the print processing method have been described on assumption that intermediate data are divided for each predetermined band, and that the divided intermediate data are then processed for each page. However, the embodiment is not limited to this configuration and, alternatively, the intermediate data may be processed for each page. For example, an output buffer memory may have a capacity for one page, and drawing using the edge drawing means may be performed for each page.

[0099] Moreover, in the description of the embodiment, although the intermediate data format having contour data of a trapezoid or a triangle is adopted, the intermediate data

are not limited to these. Alternatively, contour data of a convex polygon, of an arbitrary polygon, or of a figure with an arbitrary shape including a curved line may be used.

[0100] Furthermore, in the above-described embodiment, edge information generated by the edge information generation means contains a start point, an end point and gradation data of a filling line parallel to a scanning line of the print output means. However, the edge information generation means may generate edge information which contains a start point, an end point and gradation data of a filling line parallel to a sub-scanning line of the print output means. Alternatively, the edge information generation means may generate both filling lines parallel respectively to a scanning line and a sub-scanning line of the print output means. In these cases, the gradation conversion means and the edge drawing means respectively perform gradation conversion and bitmap drawing of a corresponding area on the basis of the generated edge information.

[0101]

[Effect of the Invention] As described above, in the print processing apparatus of the present invention, which expands print data in PDL or the like in bitmap, and which then outputs the expanded print data, an edge data generation unit, a binarization processing unit, and an edge drawing unit are configured of hardware. Moreover, the generating of edge data, the binarization process of an area defined by the edge data, and the drawing process of the binarized data to the output buffer, are performed in parallel and in pipeline.

Thereby, faster printing can be achieved than otherwise.

[0102] Furthermore, the entirety of the print processing apparatus is configured so that the binarization process is performed on a drawing object, on an edge-by-edge basis, in synchronization with the drawing process, immediately before drawing the data to a buffer memory for output. This configuration makes it possible to eliminate the necessity of a large-capacity page memory or a large-capacity band memory, in which multi-valued gradation data are temporarily stored in the conventional manner. As a result, it is possible to provide a print processing apparatus and a print processing method with which fast printing can be achieved at low costs.

[Brief Description of the Drawings]

[Figure. 1] Fig. 1 is a diagram showing a basic configuration of a print processing apparatus of the present invention.

[Figure. 2] Fig. 2 is a block diagram showing a basic configuration of an embodiment of the print processing apparatus of the present invention.

[Figure. 3] Fig. 3 is a conceptual diagram showing contour data of a character or a figure.

[Figure. 4] Fig. 4 is a conceptual diagram showing a process in which edge data are generated from intermediate data.

[Figure. 5] Fig. 5 is a diagram showing a state in which threshold matrix data M are stored.

[Figure. 6] Fig. 6 is a block diagram showing an internal configuration of a binarization processing unit of the print processing apparatus of the present invention.

[Figure. 7] Fig. 7 is a flowchart showing a flow of an intermediate data generation process of the print processing apparatus of the present invention.

[Figure. 8] Fig. 8 is a flowchart showing the flow of an edge data generation process of the print processing apparatus of the present invention.

[Figure. 9] Fig. 9 is a flowchart showing the flow of an edge drawing process of the print processing apparatus of the present invention.

[Figure. 10] Fig. 10 is a flowchart showing the flow of a binarization process of the print processing apparatus of the present invention.

[Figure. 11] Fig. 11 is a view illustrating an example of a state in which objects overlap each other.

[Figure. 12] Fig. 12 is a flowchart showing an example of a binarization process including a correction process which is performed on edge data by considering an overlap of objects.

[Figure. 13] Fig. 13 is a conceptual diagram of a binarization process performed by a dither method.

[Figure. 14] Fig. 14 is a conceptual diagram of a binarization process performed by a density pattern method.

[Figure. 15] Fig. 15 is a conceptual diagram of a binarization process in a conventional print processing apparatus.

[Figure. 16] Fig. 16 is a diagram illustrating a print information processing process of the present invention.

[Figure. 17] Fig. 17 is a diagram illustrating examples of intermediated data and edge data in a print information processing apparatus of the present invention.

[Description of Reference Numerals]

1 PRINT DATA; 2 EDGE DATA GENERATION MEANS; 3 BINARIZATION MEANS; 4 EDGE DRAWING MEANS; 5 PRINT OUTPUT MEANS; 11 PRINT DATA SPOOL UNIT; 12 PRINT DATA INTERPRETATION UNIT; 13 INTERMEDIATE DATA GENERATION UNIT; 14 INTERMEDIATE DATA STORING UNIT; 15 INTERMEDIATE DATA INPUT UNIT; 16 EDGE DATA GENERATION UNIT; 17 BINARIZATION PROCESSING UNIT; 18 THRESHOLD DATA STORING UNIT; 19 EDGE DRAWING UNIT; 20 OUTPUT DATA STORING UNIT; 21 PRINTING DEVICE CONTROL UNIT; 22 PRINTING DEVICE

Continued from the front page

F Term (Reference)	2C061	AP01	HH03	HJ06	HM07	
	2C087	BA02	BA03	BA04	BA05	BA12
		BC05	BD05	BD24	BD41	
	5B021	CC05	LG08			
	5B080	FA05	FA07			
	5C077	LL18	MP05	MP07	NN09	PP00
		PP15	PP20	PP47	PP68	PQ08
		PQ20	PQ22	PQ23	PR02	TT03
		TT06				

Drawings

Fig. 1

1 PRINT DATA

2 EDGE DATA GENERATION MEANS

3 BINARIZATION MEANS

4 EDGE DRAWING MEANS

5 PRINT OUTPUT MEANS

FIG. 2

1 PRINT DATA

11 PRINT DATA SPOOL UNIT

12 PRINT DATA INTERPRETATION UNIT

13 INTERMEDIATE DATA GENERATION UNIT

14 INTERMEDIATE DATA STORING UNIT

15 INTERMEDIATE DATA INPUT UNIT

16 EDGE DATA GENERATION UNIT

17 BINARIZATION PROCESSING UNIT

18 THRESHOLD DATA STORING UNIT

19 EDGE DRAWING UNIT

20 OUTPUT DATA STORING UNIT

21 PRINTING DEVICE CONTROL UNIT

22 PRINTING DEVICE

Fig. 3

BAND

Fig. 5

ADDRESS A

THRESHOLD MATRIX M

Fig. 6

GRADATION DATA

COORDINATES DATA

18 THRESHOLD DATA STORING UNIT

171 BIT SHAPING UNIT

172 MEMORY CONTROL UNIT
173 COMPARING UNIT
FILLING PATTERN

Fig. 7
INTERMEDIATE DATA GENERATION PROCESSING START
S1 INPUT PRINT DATA
S2 INTERPRET PRINT DATA
S3 IS INPUT DATA DRAWING COMMAND?
S4 IS INPUT DATA DRAWING STATE COMMAND?
S5 SET ATTRIBUTE AND THE LIKE
S6 GENERATE INTERMEDIATE DATA
S7 IS ONE PAGE PROCESSED?
S8 DIVIDE BAND

Fig. 8
EDGE DATA GENERATION PROCESS START
S11 INPUT INTERMEDIATE DATA
S12 CALCULATE DDA INITIAL VALUES
S13 OUTPUT EDGE DATA
S14 IS OBJECT COMPLETED?
S15 CALCULATE DDA
S16 IS BAND COMPLETED?

Fig. 9
EDGE DRAWING PROCESS START
S21 INPUT EDGE DATA
S22 PERFORM BINARIZATION PROCESS
S23 TRANSFER BITMAP DATA FOR ONE WORD
S24 MEMORY ACCESS
S25 IS EDGE COMPLETED?

Fig. 10
BINARIZATION PROCESS START
S31 INPUT GRADATION DATA
S32 INPUT THRESHOLD DATA

S33 PERFORM COMPARING PROCESS
S34 OUTPUT BITMAP

Fig. 11
OBJECT A
OBJECT B

Fig. 12
S41 DETERMINE OVERLAPPING OF OBJECTS
S42 CORRECT EDGE DATA
S43 INPUT CORRECTED EDGE DATA
S44 PERFORM BINARIZATION PROCESS

Fig. 13
INPUT DATA
THRESHOLD DATA
OUTPUT DATA

Fig. 14
INPUT DATA
THRESHOLD DATA
OUTPUT DATA

Fig. 15
PDL EXPANSION MEANS
PAGE MEMORY
BINARIZATION MEANS
BITMAP DATA

Fig. 16
PRINT DATA
13 INTERMEDIATE DATA GENERATION UNIT
16 EDGE DATA GENERATION UNIT
17 BINARIZATION PROCESSING UNIT
BITMAP DATA

Fig. 17
INTERMEDIATE DATA 100
EDGE DATA 200